Mouse 2.0: Multi-Touch Meets the Mouse

Nicolas Villar[1], Shahram Izadi[1], Hrvoje Benko[2], John Helmes[1], Dan Rosenfeld[3]
Eyal Ofek[3], Jonathan Westhues[3], Alex Butler[1], Xiang Cao[1], Bill Chen[3], Steve Hodges[1]

[1]Microsoft Research Cambridge, 7 JJ Thomson Avenue, Cambridge CB3 0FB, UK
[2]Microsoft Research, 1 Microsoft Way, Redmond, WA, USA
[3]Microsoft Corporation, 1 Microsoft Way, Redmond WA, USA

{nvillar, shahrami, benko, jhelmes, danr, eyalofek, jonawest, dab, xiangc, bilchen, shodges}@microsoft.com

Figure 1. Our multi-touch mice explore different touch sensing techniques, form-factors and interactive affordances.

ABSTRACT

In this paper we explore the possibilities for augmenting the standard computer mouse with multi-touch capabilities so that it can sense the position of the user's fingers and thereby complement traditional pointer-based desktop interactions with touch and gestures. We present five different multi-touch mouse implementations, each of which explores a different touch sensing strategy, which leads to differing form-factors and hence interaction possibilities. In addition to the detailed description of hardware and software implementations of our prototypes, we discuss the relative strengths, limitations and affordances of these different input devices as informed by the results of a preliminary user study.

ACM Classification: H.5.2 [Information interfaces and presentation]: User Interfaces. – Input devices and strategies; Graphical user interfaces.

General terms: Design**,** Human Factors

Keywords: Multi-touch, mouse, surface computing, desktop computing, novel hardware, input device.

INTRODUCTION

Humans are naturally dexterous and use their fingers and thumbs to perform a variety of complex interactions with everyday objects to a high precision. The traditional computer mouse design, however, makes little use of this dexterity, reducing our hands to a single cursor on the screen. Our fingers are often relegated to performing relatively simple actions such as clicking the buttons or rolling the mouse wheel.

With the emergence of multi-touch, we now have the opportunity to manipulate digital content with increased dexterity. But whilst multi-touch has been incorporated into many different form-factors from tabletop to mobile phone, it has yet to find a place on our desktops. Indeed, many multi-touch systems have been specifically motivated by moving computing 'away from the desktop'. Nonetheless, the traditional desktop computer comprising of vertical display, keyboard and mouse still dominates, remaining an ergonomically comfortable and practical configuration for carrying out many computing tasks.

In this paper, we explore the challenges of bringing the benefits of multi-touch interaction to a traditional desktop computing setting, where the primary pointing device is a mouse. Given the domination of the mouse on the desktop, it is hard to contemplate replacing it with a different device in order to support multi-touch interaction. Instead, we feel there is an opportunity to extend the basic mouse with multi-touch capabilities and we refer to these extended mice as *multi-touch (MT) mice*. In addition to serving as traditional 2D point-and-click devices, MT mice conceptually allow the user to reach into the GUI – enabling them to manipulate the graphical environment with their fingers, and to execute commands via hand-gestures without the need to physically touch the display.

The main contribution of this paper is a technical exploration of the design space for MT mice through five different hardware prototypes. After reviewing related work in this area, we present a detailed description of the technical implementation of each of our MT mice prototypes, focusing first on the hardware and then on the low-level sensing and processing. We go on to describe how the MT mouse sensor data is used to enhance the desktop user experience, and

UIST'09, October 4–7, 2008, Victoria, BC, Canada.

conclude by discussing the possibilities afforded by the different mice implementations and their relative merits, as informed by the results of a pilot user study.

RELATED WORK

The basic computer mouse design has remained essentially unchanged for over 40 years following its first public demonstration by Doug Englebart et al. [9]. Since then, repeated efforts have been made to augment the basic mouse functionality with additional capabilities. Arguably, the most successful addition has been the scroll wheel [30] which was originally added to support 3D interactions.

One of the primary areas of research in this space has focused on extending the number of degrees of freedom (DOF) that the mouse can sense and thereby control. MacKenzie et al. [19] and Fallman et al. [10] describe prototype devices that contain hardware from two mice rigidly linked into a single chassis to enable rotation sensing and thereby provide 3DOF input. Rockin'Mouse [2] augments a mouse with tilt sensors to enable 4DOF input. The bottom of the device is rounded to facilitate this rocking motion, which is used to control two extra degrees of freedom suitable for manipulation of 3D environments. VideoMouse [14] is a mouse augmented with a camera on its underside and employs a mouse pad printed with a special 2D grid pattern. It uses computer vision to detect changes in the grid pattern to support full 6DOF input, including tilt, rotation and limited height sensing. Manipulating the mouse in mid-air is also possible with mice that include accelerometers and gyroscopes (e.g., [1][12]).

Cechanowicz et al. [7] investigated the use of uni- and dual-pressure augmented mice, where one or more pressure sensors mounted on the mouse simultaneously control cursor position as well as multiple levels of discrete selection for common desktop applications. Kim et al. [17] investigated the concept of an inflatable mouse which could also be used for pressure sensitive input.

PadMouse [3] adds a touchpad on top of the mouse. This single-touch sensing prototype demonstrates the benefits of such a configuration in precise pointing tasks. Similar benefits can be achieved by substituting the absolute position-sensing touchpad for a relative-position sensing mini joystick (e.g. TrackPoint Mouse [29]) or a miniature trackball (e.g. MightyMouse [23]). In contrast to our work, these approaches only support single fingertip input.

While not allowing for multi-touch interactions on a single mouse device, Latulipe et al. [18] have investigated symmetric bimanual input performed with two mice in conjunction with a desktop display, finding this superior to the asymmetric case or using a single mouse. Absolute sensing of the mouse location on the surface has been explored in the FieldMouse project [28].

Our work also draws inspiration from numerous interactive surface products and prototypes which enable multi-touch interactions through either capacitive (e.g., [8][26]) or camera-based methods (e.g., [21][13][22]). Forlines et al. [11] evaluated the benefits of direct touch vs. standard mouse input for interactive tabletops and found overall preference for direct touch, but noted that mouse input might be more appropriate for standard applications requiring precise single-point input.

There are of course other ways to bring multi-touch interactions to the desktop, rather than augmenting the mouse. For example, it is possible to augment the vertical display with direct input capabilities. There have been several attempts to mitigate the resulting precision and occlusion problems [27], for example using bimanual multi-touch interactions [5] or placing the contacts behind the screen [33]. However, this is still not the most ergonomic configuration for desktop use – user's hands and arms will quickly fatigue and users have to explicitly switch between using the touchscreen and the mouse for input.

The benefits of multi-touch interactions can also be achieved with multi-touch sensitive pads that are not coupled with the display (e.g., [15] or the touchpad in Apple laptops). Malik et al. have also [20] explored how camera-tracked multi-finger gestural touchpads can be used for indirect input to large displays. Moscovich et al. have developed a number of multi-finger interaction techniques and graphical cursors for multi-touch pads [25]. However, as surveys have revealed [16], most users prefer mice to touchpads, especially for precise selection tasks.

Given the limitations of touch screens and pads for bringing multi-touch to the desktop, we explore a different approach for enabling such capabilities, which takes the mouse as the starting point. The mouse has gone through several decades of iterative refinement; it offers high resolution pointing, is ergonomically designed to be held in a single hand and requires little effort to use. It is a notoriously well-established device for desktop and we feel that there are opportunities for complementing the capabilities of regular mice with the compelling new interactions afforded by multi-touch systems.

HARDWARE DESIGN PROTOTYPES

This paper focuses on the technical exploration of the design space for MT mice, and aims to broaden understanding of how to build these novel input devices. This is a new area of research, and we believe that understanding how to develop such devices in the face of many technical challenges is a key starting point. In this section we present five MT mouse hardware devices; each presents a different implementation and sensing strategy that leads to varying device affordances and form-factors and hence very unique interaction experiences.

One of our main goals when realizing these mice is to support multi-touch gestures *alongside* regular mousing operations. MT mice should therefore still allow the user to easily *grasp* and *release* the device, *move* it with their wrist or forearm, *clutch* it for repositioning, and perform standard cursor interactions such as *clicking*, *dragging* and *selection* without compromising precision.

We describe each prototype in turn, explaining our motivations and rationales behind each effort, and outlining on the hardware design and key implementation details.

FTIR Mouse

Our first MT mouse design is based on a common technique for enabling multi-touch input on interactive surfaces: frustrated total internal reflection (FTIR) [13]. With this approach a sheet of acrylic is edge-lit with infrared (IR) light. When a finger is pressed up against the acrylic, it causes IR light to be scattered away from the finger; this can be detected using an IR camera which is imaging the surface. Although this technique has been used to provide multi-touch for a variety of systems, our approach is novel in that it applies FTIR to the surface of an indirect input device, augmented with a regular mouse sensor. Our FTIR Mouse is shown in Figure 2.

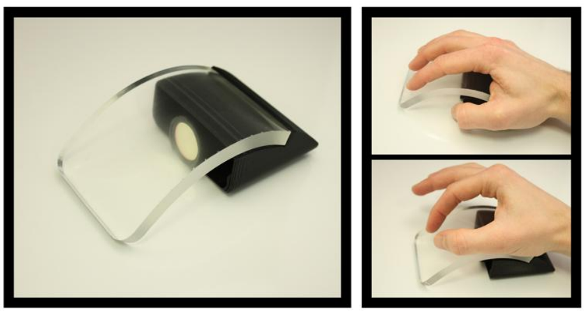

Figure 2. FTIR Mouse applies the principle of frustrated total internal reflection to illuminate a user's fingers, and uses a camera to track multiple points of touch on its curved translucent surface.

In order to adapt this technique into a form-factor suitable for a mouse, we molded a piece of acrylic into a smooth arc shape. The acrylic arc is mounted into a custom base containing a row of IR LEDs, in such a way that the edge of the arc is pressed flush against the LEDs. The base also contains a standard optical mouse sensor to track its displacement across the surface, as well as a small PGR FireFly MV camera equipped with a wide-angle lens, mounted so that it captures the underside of the arc in its entirety.

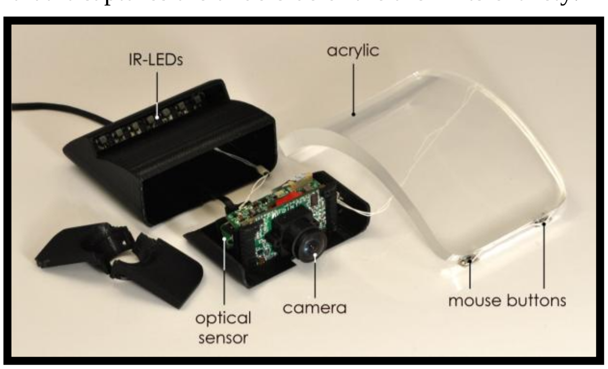

Figure 3. The main components of FTIR mouse.

Small buttons mounted under the forward edge of the arc are linked to the internal mouse circuitry, allowing the user to perform mouse-clicks by depressing the front of the device with their fingers. Figure 3 shows the main components of the FTIR mouse.

Figure 4 shows an example image captured by the camera when a user touches the front surface of the arc with three fingers. These touching fingers are clearly illuminated as they touch the surface because they cause some of the IR light to scatter to the camera. These images are processed using a vision pipeline described later, to derive the position of touching fingers.

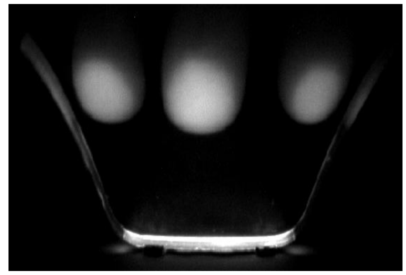

Figure 4. The IR camera with wide angle lens captures the front of the FTIR mouse. In this example, the three fingers touching are illuminated.

We found the FTIR approach to be a suitable technique for prototyping fingertip MT tracking on the surface of the mouse. FTIR inherently gives a binary indication of when the user is touching the surface which makes the interaction robust in operation. Also, from an industrial design perspective, the use of the clear acrylic affords some interesting aesthetic possibilities. However, the FTIR technique does have limitations as a means of sensing multi-touch, and places some restrictions on the physical design of the device (which may be at odds with ergonomic requirements). For example, sensing is limited to the area at the front of the device (in the camera's field of view), meaning that only the user's outstretched fingertips can be sensed. The use of an IR-sensitive camera as a sensor makes the device susceptible to sunlight and other external sources of IR light – a well-known problem of FTIR-based interactive surfaces. Furthermore, the shape and curvature of the transparent acrylic section cannot be chosen arbitrarily, as steep curves or a convex outline would break the total internal reflection. In order to address some of these limitations, our next prototype explores an alternative hardware implementation: the use of *diffuse* IR illumination to track a user's hands on a surface, coupled with additional optics which extend the field of view of the camera.

Orb Mouse

Orb Mouse is shown in Figure 5. It facilitates multi-touch sensing on its hemispherical surface by incorporating an IR-sensitive camera and internal source of IR illumination. Unlike FTIR Mouse, the illumination is not totally internally reflected through the shell of the device; rather, it radiates outwards from the centre of the device, and is reflected back into the camera by objects (such as the user's hands) that come into close proximity to the hemispherical surface of the mouse.

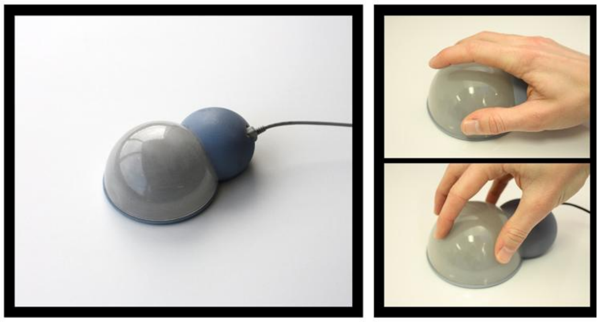

Figure 5. Orb Mouse is equipped with an internal camera and a source of diffuse IR illumination, allowing it to track the user's hand on its hemispherical surface.

The basic principle of operation is similar to the numerous interactive surface technologies that depend on diffused IR illumination (e.g., [21]). Figure 6 illustrates the internal construction of our prototype. We again use a PGR FireFly MV camera with an IR-pass filter, together with 4 wide-angle IR LEDs as the illumination source. Instead of pointing directly at the surface, the camera is aimed towards an internally mounted hemispherical mirror. This gives the camera a very wide angle view of most of the mouse surface. Folding the optics in this way also has the significant benefit of maintaining a relatively low-profile form factor which is critical if the device is to be used in a mouse-like manner.

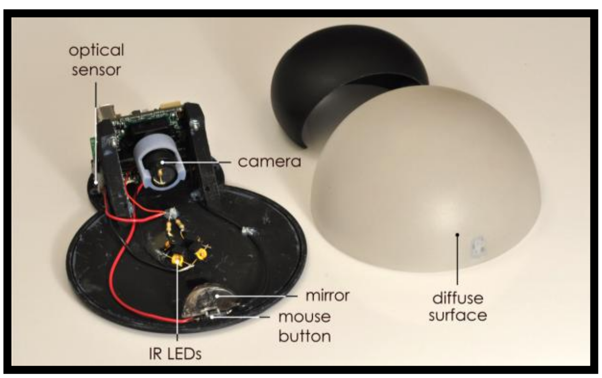

Figure 6. The main components of Orb Mouse.

In addition to the MT-sensing components, standard optical mouse circuitry is integrated to the base of the device to track its displacement across the surface and a microswitch is placed underneath the mirror to allow the entire hemisphere to be "clicked".

As shown in Figure 7 left, the camera image of the reflector is heavily distorted; much more so than with FTIR mouse. We undistort this image using a technique similar to that described in [4]. This image is further normalized (to account for non-uniform illumination across the surface), binarized, and finally a connected component analysis is used to locate the centre of any touching contacts. This pipeline is highlighted in Figure 7. Note that FTIR mouse uses a similar vision pipeline, with the difference that the image is initially undistorted due to the simpler optical geometry.

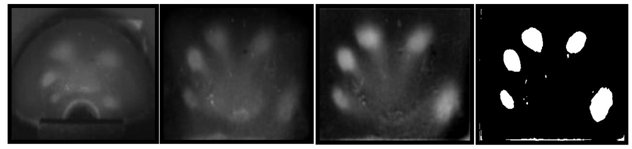

Figure 7. The vision pipeline for Orb Mouse. The image is captured from the mirror, undistorted, normalized and binarized before the position of individual contacts are calculated (from left to right).

The hemispherical shape of Orb Mouse is intended to be relatively easy to grip and the constant curvature ensures that the user's fingers experience a smooth gradient while moving from side to side and front to back (similar to the interactions on a flat surface). In addition, Orb Mouse's actively-sensed interaction area is substantially larger than that of FTIR Mouse, encompassing both the top and sides of the devices, thereby allowing all fingers and even the whole hand to be simultaneously engaged in interactions.

As with the FTIR Mouse design, the Orb Mouse is sensitive to IR light from external sources. Although the use of diffuse illumination coupled with folded optics affords greater flexibility in form-factor, it is also more noisy and susceptible to interference; the reflected-IR images of the user's touch-points, as captured by the camera, are considerably lower contrast than those possible with an FTIR design. To overcome some of these issues, we have also explored alternatives to camera-based sensors, described in the next section.

Cap Mouse

The Cap Mouse prototype tracks the position of multiple fingers on its surface through capacitive touch sensing as shown in Figures 8 and 9. This prototype uses a flexible matrix of capacitive-sensing electrodes to track the location of the user's contacts.

Figure 8. Cap Mouse employs a matrix of capacitive touch-sensing electrodes to track the position of the user's fingertips over its surface.

In contrast to previous designs which use capacitive sensing for detecting clicks only [23], 1D scrolling [1], or the single finger position [3], the Cap Mouse design is novel in that the device includes a true multi-touch sensor and thus is able to simultaneously track the locations of all of the user's fingers on the surface of the mouse. In addition to capacitive multi-touch sensing, the base of the mouse contains a regular mouse sensor and single mouse button which can be clicked by pressing down towards the front of the device.

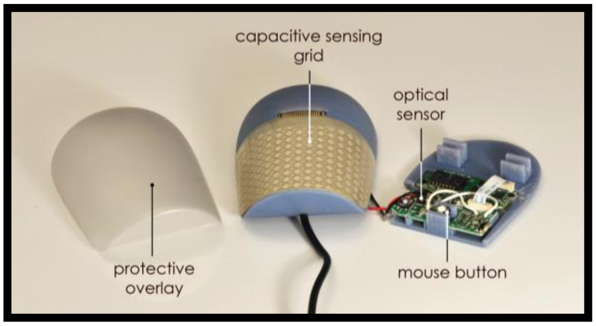

Figure 9. The main components of Cap Mouse.

Figure 9 illustrates the internal components of our prototype. An X-Y grid of sensing elements is printed on a flexible plastic substrate using conductive ink. This sensor is wrapped around a portion of the mouse's surface and covered with a thin plastic shell to prevent direct electrical contact between fingers and the sensor elements. When a user's finger is placed on the shell it affects the mutual capacitance between the sensing elements of nearby rows and columns. This can be detected and pinpointed using a microcontroller which sequentially scans the various combinations of rows and columns.

Capacitive sensor elements are placed with center-to-center spacing of around 5 mm but the effective resolution of the sensor is actually much finer than this – a finger will generally cover multiple sensor elements so it is possible to interpolate position from multiple capacitance readings.

The raw capacitive sensor values are converted into a 20x10 grayscale image, which is interpolated up by a factor of 10 along each axis. This image is then fed through part of the same vision pipeline as FTIR and Orb Mice, from binarization onwards, to extract out the location of touching fingers.

The capacitive approach is an appealing means of constructing a surface touch sensor. Unlike our optical mice, it is immune to ambient illumination. The sensor provides much less data than the cameras included in other designs—thus lowering bandwidth and processing requirements—while still allowing good positional precision through interpolation. Cap mouse is also physically more compact because the design constraint imposed by the optical path required in our vision-based prototypes is eliminated. The compactness of the sensor enabled us to design a mouse with a relatively conventional form and scale, and thus investigate the pros and cons of performing multi-touch gestures on an otherwise normal mouse. It also has relatively low power consumption. However, the effective resolution of the capacitive sensor is considerably lower than with a camera-based approach.

Side Mouse

In contrast to the previous three prototypes which augmented the surface of a mouse with multi-touch sensing capabilities, the Side Mouse device senses the user's fingers as they move across the surface of the desk around the periphery of the device. This design is inspired by [6] which explored proximity based sensing around the periphery of mobile devices. In contrast to the other prototypes described above, Side Mouse veers away from conventional mouse form-factors, and is designed to sit under the user's palm, leaving the fingers free to touch the surface around the device as shown in Figure 10.

Figure 10. Side Mouse is augmented with a proximity-sensing technique called SideSight, which enables multi-finger sensing on the surface around the device.

The key components of the device are highlighted in Figure 11. The base is equipped with a forward-sensing camera, mounted behind an IR-pass filter. Underneath the camera, and suspended a few millimeters above the surface, sits a line-generating IR-laser illuminator which casts a sheet of IR light that fans outwards from the front of the device. An ultra-wide angle lens allows the camera to image the area covered by the illuminator. Fingers and other objects placed in this area reflect IR light back to the camera, allowing it to sense their positions as shown in Figure 12. These images are once again processed using the same vision pipeline presented earlier.

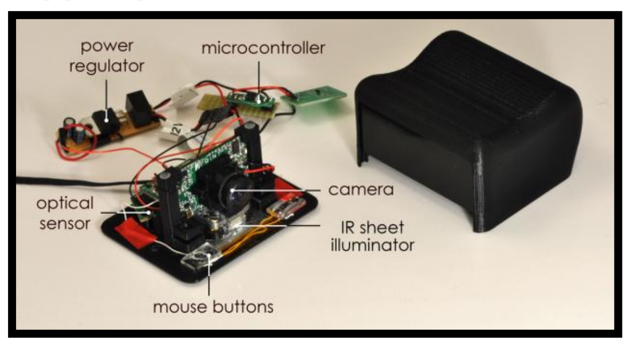

Figure 11. The main components of Side Mouse.

In addition, the base of the device is equipped with an optical mouse sensor allowing it to carry out regular pointing tasks. Since the user's fingers rest directly on the surface, performing a mouse-click with this device is achieved by pressing it down with the palm of the hand. This action is detected by a pair of buttons mounted on the underside of the case, which require an actuation force high enough that they are not triggered accidentally when simply resting a palm on the device, yet not so high to require an undue amount of force to perform a mouse click.

Figure 12. The Side Mouse camera image. Two fingers are hovering above the surface and are not illuminated (left). Once touching the surface, the IR beam is broken by the fingers and these become brightly illuminated (right).

The key interaction possibility that we explore with Side Mouse is the ability to create a multi-touch area that is not restricted to the physical surface of the device's casing. As well as interactions whilst the device is 'gripped', the main body of the mouse can also be 'parked' – that is, moved to a location and subsequently released – to define an ad-hoc interactive region on the surface of the desk. This wide sensing scope does however have practical implications in detecting stray objects (not just fingers) like the keyboard and other items on the desk.

Arty Mouse

Side Mouse opens up the interaction space around the mouse. However, like our other camera mice it has issues regarding susceptibility to lighting and higher power consumption. Our final prototype, which we call Arty Mouse (short for *articulated* mouse), takes the notion of Side Mouse one step further.

Figure 13. Arty Mouse is equipped with three high-resolution optical mouse sensors: one in the base, which rests under the user's palm, and two under the articulated extensions that follow the movements of the index finger and thumb.

In our Arty Mouse design (shown in Figure 13), the palm of the hand rests on the base of the device; from this base extend two articulated 'arms' that can be freely and independently moved two-dimensionally by the thumb and index finger. The design makes use of three separate optical mouse sensors – one under the base and one underneath each articulated arm – to individually track the displacement of each of these parts across the surface. The design is tailored towards use with the thumb and index finger, although other finger arrangements are also possible.

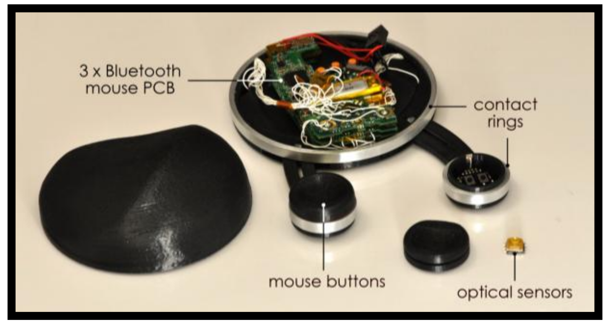

Figure 14. The main components of Arty Mouse.

The base of the Arty Mouse houses the circuitry from three separate Bluetooth optical mice as shown in Figure 14, making this our only wireless MT mouse currently. For our prototype we chose to use components extracted from MoGo mice [24], due to their extremely thin form factor. The optical sensors on these devices were decoupled from their original circuitry and re-soldered it to a small (2cm diameter) PCB of our own design, which includes some passive components necessary for correct operation. One of the sensors is placed on the underside of the base, and the other two at the end of the articulated arms.

The arms are attached to the base using a grooved pivot mechanism, allowing them to be moved with 2DOF while still retaining some desirable physical constraints that help maintain mechanical stability while moving and clutching the device. In addition to the mouse sensor, each arm is equipped with a button that can be depressed with the fingertips. Conductive metal rings surround the base of the device as well as the extremities of the arms. These act as contact sensors, detecting whenever two of these parts (the base and index part, the base and thumb part, or the thumb and index part) are touching.

It is important to note that the mice sensors are *relative position* devices, but for some applications *absolute position* of each part with respect to each other may be desired. This could be achieved using additional sensors on the articulated arms – such as rotary and linear encoders or potentiometers – to estimate their angle and distance from the base. However, for the sake of simplicity in an already complex mechanical design we opted instead for a dead-reckoning software technique, where the relative movement vectors of the arms are summed to estimate their current position with respect to the base. With this technique it is important to establish a ground-truth (a known position from which to start measuring changes in position), and for this we bring into play the metallic contact rings: when the base, index or thumb touch each other, which happens regularly and naturally during interaction with the device, this provides an indication of their absolute position along one axis.

One key advantage of this particular design over other sensing techniques explored in this paper is the fact that it allows a high-resolution optical mouse sensor to be placed underneath two of the user's fingers. This technique provides extremely high sensing fidelity compared with capacitive or camera-based sensing techniques described earlier, and can be leveraged to support subtle and fine grained multi-touch gestures.

COMBINING MOUSE INPUT WITH MULTI-TOUCH

Enriching the Cursor with the MT Cloud

In order to accommodate multi-touch interaction - while still supporting traditional mousing techniques - we have developed the Multi-Touch Cloud cursor, which is an augmented version of the traditional GUI mouse cursor.

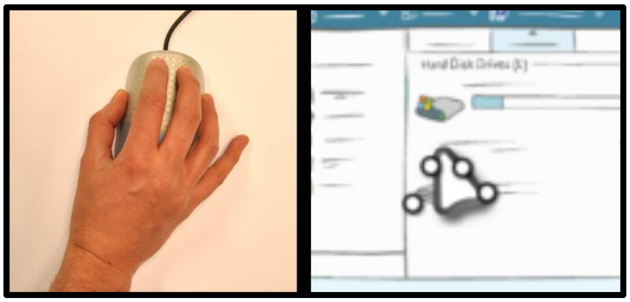

Figure 15. Multi-touch points on the mouse are mapped to a small region around the cursor, which we call the *MT Cloud*. Each touch point is visualized as a small dot around the cursor.

As with a regular cursor, the on-screen position of MT Cloud cursor is driven by the displacement of the mouse on the surface. In addition, the cursor is surrounded with a number of small points, which appear when a finger-touch point is detected by the MT sensing capabilities of the device (see Figure 15). Each dot corresponds to a touch point, and the position of the dots relative to the center of the cursor maps to the position of the fingers on the surface of the device (or on the periphery of the device, in the case of Side Mouse and Arty).

The dots provide visual feedback to the user about the sensed position of their fingers, obviating the need to look at the hand or device during interaction. Additional input coordinates are generated at the location of each of these MT points, and are injected into the input queue for use by compatible graphical elements that can receive MT events in addition to regular mouse events. This allows the user to, for example, select an onscreen object with the mouse cursor, and perform MT manipulations that are limited to that particular target in a way that maintains the notion of input focus that is familiar to users of graphical interfaces.

The multi-touch points are bounded within a relatively small point cloud around the mouse cursor. The dimensions of the cloud area - and the gain function associated with their movement - varies between each of our five devices to compensate for differences between the individual physical interaction areas and sensing fidelity.

Multi-touch Applications on the Desktop

One initial goal we had was to run existing MT applications on the desktop, both to validate our MT mouse implementations and to further unpack the utility of MT on the desktop. In so doing we have built a generic bridge between the Microsoft Surface SDK [22] and our MT devices. This allows each of our devices to pipe their MT data into regular Surface applications using an implementation of the MT cloud technique. We support standard multi-touch gestures for rotating, scaling and flicking of on-screen objects, without compromising standard single cursor drag-and-drop and selection operations. By using the $1 Gesture Recognizer technique [32], we have also implemented a way to associate and trigger application functionality using simple gestures that can be performed on the touch-sensing areas with a single finger.

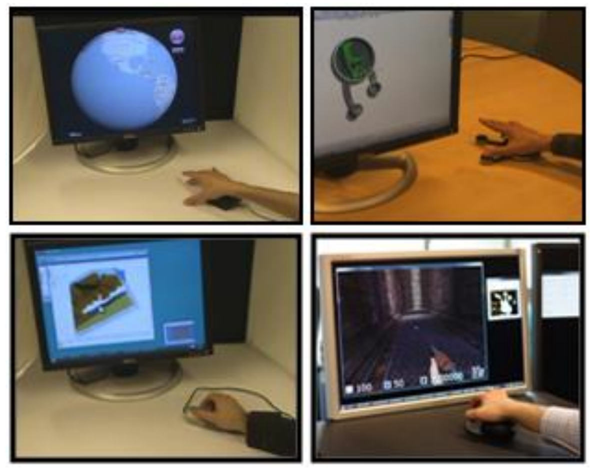

Figure 16. Applications running with our MT mice. Clockwise from top-left: manipulating Virtual Earth, precise modeling in SolidWorks, controlling a FPS game and browsing using a desktop mockup.

We have only begun to explore the use of these devices for specific applications, which will be the focus for our future work. One interesting possibility is the ability to map the additional degrees of freedom that are opened up by multi-touch to more intuitive 3D manipulations. We have demonstrated the use of Arty to allow both cursor control and 3D camera manipulation in the SolidWorks CAD application using the additional articulated sensors, each offering an additional 2DOF. We have also explored mapping the rich MT data sensed from Orb Mouse to a first person shooter game, allowing simultaneous control of the virtual camera in 6DOF and other controls that would typically be associated with keyboard shortcuts, such as changing weapons. These are shown in Figure 16.

PILOT STUDY

In order to better understand the affordances of each device, and get an initial sense of their relative strengths and limitations, we conducted a pilot user study. We asked a group of 6 users to repeat a structured task, namely using each of our devices in turn to translate, scale, and rotate a randomly placed image to approximately match a target frame. This was based on the MT cloud technique for hybrid MT-cursor interaction.

At this stage of the work, given the broad questions surrounding the ergonomics and capabilities of our devices, we chose to focus on the qualitative aspects of the user experience, as opposed to generating quantitative results. We discuss the observations from these early-stage studies and some broader questions about the design of MT mice in the following sections.

Each user tried each of the 5 devices in sequence. After the user finished with each one, we conducted a brief interview. The user was also asked to rate the device in terms of general feeling, physical comfort, interaction intuitiveness and ease of use in comparison to other devices that had been used. The users were encouraged to think aloud during the whole process. We directly observed and video-recorded the users' behaviors when using the devices. Six volunteers, 5 right-handed and one left-handed, participated in the evaluation. These included both people with little previous experience with multi-touch input and those who have used it extensively.

Observations

All participants were able to use the 5 devices, and managed to complete the task in relatively short times. The MT cloud model seemed understandable. For example, users intuitively moved the mouse over the target before performing MT actions, if the cursor moved slightly off the target and their MT actions were ignored, they quickly noticed and moved the cursor back on target. This indicated that our users had little problem adapting to this hybrid model for input, which brings notions of cursor input and MT together.

Arty was received the most positively by the users in terms of general feeling, physical comfort and ease of use. In many ways, this is understandable given that the two articulated points on Arty sit comfortably under the thumb and index finger, making it extremely easy to carry out pinch gestures. We observed users very naturally carrying out these actions. The high precision of the sensors underneath each articulated point made it a very accurate device for the task. Users were able to simultaneously carry out translation, rotation, and scaling, which was not the case for any of the other devices, and this coupled with the high accuracy led to rapid and fine control. Responses in the questionnaire also highlighted that the articulated points added to the users comfort. It is interesting to note that, while Arty only supports two points of multi-touch input, the finger and thumb seemed sufficient to exercise most of the multi-touch gestures we considered.

Interestingly, Orb Mouse was also a very popular choice. Users found the device's form-factor and affordances led naturally to translation, rotation and scaling. However, rather than using a pinch gesture to perform scale and rotate, all users found that rotation was more comfortable being performed using all five fingers to grip the device and rotating these left and right in a lateral motion. For scaling up, users placed all five fingers on the top of the device and moved these down towards the base of the device (and vice-versa, when scaling down). These interactions make full use of the device's 3D shape and map well to its hemispherical form. Unlike most of the other devices, we saw the most apparent learning curve in using this device, as users 'got to grips' with this new technique for rotate and scale

Users found the concept of Side Mouse compelling, but struggled with the form-factor of the current implementation. Given the diversity of hand sizes, the device was too tall for smaller hands to touch the surface whilst simultaneously resting the device under their wrist. Users with larger hands, often found their fingers were 'out of range' of the sensor whilst their palm was resting on the device. This led to fairly bi-modal use of the device – it was typically gripped one way for mousing and then the grip was changed to perform multi-touch input. The other problem was that users felt it was uncomfortable to activate the clicker in the base of the device whilst simultaneously moving the device. This suggests a 'virtual' clicker based on the multi-touch sensor data may prove more effective. None of these limitations are insurmountable, but they do highlight how critical form-factor is in realizing such MT devices.

DISCUSSION

Being Mouse-Like

One clear aspect to emerge from our study is the importance of ergonomics and form-factor. For some devices we spent a considerable time on the form of the device (e.g. Arty), while for others the form is merely a byproduct of the technology used (e.g. Side Mouse); this was reflected in users' experiences with the device.

While there is clearly more work to be done in regards to form-factor, one of the interesting findings from our work is how receptive our users were to devices that move away from the established mouse form-factor. Initially we had hypothesized that the closer we replicate a traditional mouse shape, the more receptive users would be towards the device. We had high expectations about Cap Mouse as it was the most 'mouse-like' and would leverage users existing familiarity with such devices. However, it was often difficult for our users to shift away from this existing model when interacting with a MT capable device. This was observed on many occasions with Cap Mouse, where people would grip the device like a regular mouse, and switch to rotating objects using a virtual mouse wheel (moving the middle finger forward/backward with index finger fixed in position). Interestingly, this worked well for rotation, as gripping the mouse often led to a touch point registering as an anchor for the rotation. However, when tasked with both rotation and scaling, users struggled to perform as well and often ended up changing hand posture in order to carry out the pinch using thumb and forefinger.

The value of moving away from a mouse form-factor is also evidenced by our results regarding Orb Mouse. Here, based on interview feedback, we get the sense that seemed to users thought differently about the capabilities of the device simply because it looked and felt qualitatively different to a mouse, which led to more experimentation with MT gestures. In the case of Orb Mouse the device seemed to strike the right balance between novel form-factor and the ability to carry out regular mousing comfortably – which wasn't the case for Side Mouse, for example.

To Click or Not to Click

All our devices had physical clickers embedded in them to support standard selection tasks. We had originally considered using the clicker to explicitly 'turn on' MT gestures, but after initial testing we felt that this would be too limiting for the user. However, the notion of clicking to enable MT actually seemed intuitive to our users – they commented that being able to activate MT while not clicking 'strange'. We also observed problems leading from the need to physically press and gesture at the same time. It becomes very difficult to move fingers that are also pressing the device down, and for most devices this leads to one finger pressing down (and hence acting as a pivot for the scale or rotate) and another moving. This can be big limitation in terms of supporting more complex multi-fingered gestures. One solution might be to provide a more explicit trigger (such as a switch or dedicated touch point on the device) to enable/disable MT capabilities, although this would clearly lead to more moded interaction models.

Expose the Sensing

Another important design aspect to emerge from the study was the need to physically expose the MT sensing scope of each device. This was the most apparent for Side Mouse, where users struggled to know if fingers were within sensing range. One option specific to Side Mouse would be to use a projected laser pattern to demarcate the sensing range on the desktop. However, even for devices such as Cap Mouse where the demarcation was clearly marked, oftentimes users did not realize exactly when their fingers were being registered as touch points. This is perhaps because they rarely looked down at the mouse during desktop interactions, and so were not completely clear about where the sensing area started and ended. A bezel, such as those used on regular touch pads, could have helped by giving the user more tactile feedback. Perhaps more interestingly, we have also begun explore how this type of feedback could be provided in the UI, by way of more expressive cursor designs, such as the MT cloud described earlier. Finally, in terms of physical design of the device, it seems important to provide inactive areas where the user can comfortably rest their fingers while clutching the device without accidentally triggering the MT input.

All About Constraints

One of the main comments from users was that some of the devices provided 'too much freedom'. We had anticipated that this additional freedom would lead to more open interactions, but conversely users sometimes felt less comfortable experimenting and interacting because they simply couldn't predict how they should interact with the device. A clear exception here was Arty, whose physical form afforded particular places to rest fingers and palm, and its physical constraints clearly demarcated the range of gestures that were possible. Rather than limiting our users, they realized they were holding and interacting with the device in the manner it was designed for, and seemed comfortable to experiment within that design. Obviously users can be trained to use more open-ended devices, but this finding suggests that molding the shape of the device to suggest the ways that it might be held and interacted with might reduce its initial learning curve.

Don't Mode Me In

It also became apparent from the user study that some of our devices are inherently bi-modal in their support of MT versus regular mousing. This modal nature was particularly clear for FTIR, Cap and Side mice and led to occasional frustrations. Users would often 'posture switch', gripping the device in one way to perform MT gestures and another to perform pointing tasks. This was mainly due to the fact that the thumb and forefinger were primarily used to carry out the gestures, and that the form-factors of these devices required the thumb to be repositioned on the side of the device in order to grip it and translate it..

The bi-modal interaction led to a mostly sequential division of labour when carrying out tasks. However, and particularly with Arty mouse, we did observe cases of simultaneous interaction, where the rotation and scale of the object would be adjusted with the fingers while being translated towards the target using arm and wrist movements.

One of the interesting challenges of placing a multi-touch sensor on the surface of the mouse is that the device needs to be able to be held while in use. This is necessary both to allow the user to move the device on the surface with their wrist and arm, as well as to allow for typical mouse clutching behavior where the device is gripped and lifted of the surface in order to reposition it.

CONCLUSION

This paper has explored a number of ways of introducing multi-touch capabilities to the mouse as a way to make MT interaction more widely applicable to the desktop environment. We have begun to chart the design space for novel types of input devices that combine regular mousing capabilities with MT gestures. The core contribution of our work is a technical one – we have established the feasibility of building multi-touch mice, and have documented a variety of approaches for doing so. However, the exercise of building these prototypes has been valuable to us beyond the resulting set of devices. Through the process of design and development, we have come to experience first-hand the tension between technical challenges and ergonomic requirements that lie at the heart of making multi-touch mice practical and desirable.

More concretely, our contributions include: a practical comparison of five different techniques for enabling multi-touch on the desktop PC, which include three distinct camera-imaging approaches, the use of capacitive sensors to track multiple fingers on a curved surface, and an approach for tracking finger movements using multiple optical mouse sensors; the MT Cloud cursor, which doubles as a way to model indirect multi-touch input for GUIs and as a mechanism to provide feedback to users; and, our reflections on the general issues regarding MT mice – informed both by the insights gained from our design and development efforts, as well as through the initial user feedback from our preliminary study. In future work, we plan to refine our prototypes – both ergonomically, and in terms of their sensing capabilities – to deeper explore the interaction techniques that are specific to multi-touch enabled mice.

REFERENCES

1. Air Mouse. Logitech. http://tinyurl.com/3qtor8. (last accessed March 2009).
2. Balakrishnan, R., Baudel, T., Kurtenbach, G., and Fitzmaurice, G. (1997). The Rockin'Mouse: Integral 3D Manipulation on a Plane. *Proc. of ACM CHI '97*. p. 311–318.
3. Balakrishnan, R. and Patel, P. (1998). The PadMouse: Facilitating Selection and Spatial Positioning for the Non-Dominant Hand. *Proc. of ACM CHI '98*. p. 9–16.
4. Benko, H., Wilson, A., and Balakrishnan, R. (2008). Sphere: Multi-Touch Interactions on a Spherical Display. *Proc. of ACM UIST '08*. p. 77-86.
5. Benko, H., Wilson, A., and Baudisch, P. (2006) Precise Selection Techniques for Multi-Touch Screens. *Proc. of ACM CHI '06*. p. 1263–1272.
6. Butler, A., Izadi, S., and Hodges, S. (2008). SideSight: Multi-'Touch' Interactions around Small Devices. *Proc. of ACM UIST '08*. p. 201–204.
7. Cechanowicz, J., Irani, P., and Subramanian, S., (2007). Augmenting The Mouse With Pressure Sensitive Input. *Proc. of ACM CHI '07*. p. 1385–1394.
8. Dietz, P. and Leigh, D. (2001). DiamondTouch: A Multi-User Touch Technology. *Proc. of ACM UIST '01*. p. 219–226.
9. Engelbart, D. C. et al. (1968). A Research Center for Augmenting Human Intellect. *(demonstration)* Stanford Research Institute, Menlo Park, CA. http://sloan.stanford.edu/MouseSite/1968Demo.html.
10. Fallman, D. and Yttergren, B. (2007). The Design of a Computer Mouse Providing Three Degrees of Freedom, *HCI Internationa '07l*.
11. Forlines, C., Wigdor, D., Shen, C., and Balakrishnan, R. (2007). Direct-touch vs. mouse input for tabletop displays. *ACM CHI '07*. p. 647–656.
12. Go Pro Air Mouse. Gyration. http://tinyurl.com/l3wmow (last accessed March 2009)
13. Han, J. (2005). Low-Cost Multi-Touch Sensing Through Frustrated Total Internal Reflection. *ACM UIST '05*. p. 115-118.
14. Hinckley, K., Sinclair, M., Hanson, E., Szeliski, R., Conway, M. (1999). The VideoMouse: A Camera-Based Multi-Degree-of-Freedom Input Device. *Proc. of ACM UIST '99*. p. 103–112.
15. iGesture Pad. FingerWorks. http://www.fingerworks.com. 2009.
16. IOGEAR. Laptop Users Prefer Mice over Touchpads, Survey. PC Business Products, 2003.
17. Kim, S., Kim, H., Lee, B., Nam, T.-J., and Lee, W. (2008). Inflatable Mouse: Volume-Adjustable Mouse with Air-Pressure-Sensitive Input and Haptic Feedback. *Proc. of ACM CHI '08*. p. 211–224.
18. Latulipe, C., Kaplan, C.S., and Clarke, C.L.A. (2005). Bimanual and Unimanual Image Alignment: an Evaluation of Mouse-based Techniques. *Proc. of ACM UIST '05*. p. 123–131.
19. MacKenzie, I. S., Soukoreff, R. W., and Pal, C. (1997). A Two-Ball Mouse Affords Three Degrees of Freedom. *ACM CHI Extended Abstracts '97*. p. 303–304.
20. Malik, S., Ranjan, A., and Balakrishnan, R. (2005). Interacting with Large Displays from a Distance with Vision-tracked Multi-finger Gestural Input. *Proc. of ACM UIST '05*. p. 43-52.
21. Matsushita, N., and Rekimoto, J. (1997). HoloWall: Designing a Finger, Hand, Body, and Object Sensitive Wall. *Proc. of ACM UIST '97*. p. 209-210.
22. Microsoft Surface. Microsoft. http://www.microsoft.com/surface..(last accessed March 2009).
23. Mighty Mouse. Apple. http://www.apple.com/mightymouse/. (last accessed March 2009).
24. MoGo Mouse BT. Newton Peripherals, LLC. http://tinyurl.com/yw4lbv. 2009.
25. Moscovich, T. Hughes, J.F. Multi-finger cursor techniques. *Proc. GI '06*, 1-7, 2006.
26. Rekimoto, J. (2002). SmartSkin: an Infrastructure for Freehand Manipulation on Interactive Surfaces. *Proc. of ACM CHI '02*. p. 113-120.
27. Siek, K.A., Rogers, Y. Connelly, K.H. (2005). Fat Finger Worries: How Older and Younger Users Physically Interact with PDAs. *Proc. of INTERACT '05.* p. 267-280.
28. Siio, I., Masui, T., and Fukuchi, K. (1999) Real-World Interaction Using the FieldMouse. *Proc. of ACM UIST '99*. p. 113–119.
29. TrackPoint Mouse. (2004). IBM Research. http://www.almaden.ibm.com/cs/user/tp/tpmouse.html. (last accessed March 2009).
30. Venolia, G. (1993). Facile 3D Direct Manipulation. *Proc. of ACM CHI '93*. p.31-36.
31. Wigdor, D., Forlines, C., Baudisch, P., Barnwell, J., and Shen, C. (2007) Lucid Touch: a See-through Mobile Device. *Proc. of ACM UIST '07*. p. 269-278.
32. Wobbrock, J. O., Wilson, A.D, Li, Y. Gestures without Libraries, Toolkits or Training: A $1 Recognizer for User Interface Prototypes. *Proc of ACM UIST '07.* P.
33. Zimmerman, T., Smith, J., Paradiso, J., Allport, D. and Gershenfeld, N. Applying Electric Field Sensing to Human-Computer Interfaces. *Proc. of ACM CHI '95*. p. 280–287.